# NETLA를 이용한 二眞 神經回路網의 最適 合成

°정 종 원*, 성 상 규**, 지 석 준*, 최 우 진*, 이 준 탁***

# Optimal Synthesis of Binary Neural Network using NETLA

Jong-Won Jeong, Sang-Kyu Sung, Suk-Joon Ji, Woo-Jin Choi, Joon-Tark Lee

* 동아대학교 대학원 전기공학과
** 한국항공우주산업(주)
*** 동아대학교 전기·전자·컴퓨터 공학부

**Abstract:** This paper describes an optimal synthesis method of binary neural network(BNN) for an approximation problem of a circular region and synthetic image having four class using a newly proposed learning algorithm. Our object is to minimize the number of connections and neurons in hidden layer by using a Newly Expanded and Truncated Learning Algorithm(NETLA) based on the multilayer BNN. The synthesis method in the NETLA is based on the extension principle of Expanded and Truncated Learning (ETL) learning algorithm using the multilayer perceptron and is based on Expanded Sum of Product (ESP) as one of the boolean expression techniques. The number of the required neurons in hidden layer can be reduced and fasted for learning pattern recognition.. The superiority of this NETLA to other algorithms was proved by simulation.

**Key words:** NETLA, BNN, ETL, ESP, EBP

## 1. Introduction

N-tuple method based BNN has been used in image processing for pig evisceration and scene analysis because of having many potential advantages in knowledge manipulation.

However, the requirement problems of real-time processing in BNNs for huge quantities of data was not solved. Recently, the geometrical learning algorithm called expanded and truncated learning(ETL) algorithm was proposed to train a three layer BNN using the generation of binary to binary mapping. The constraints in ETL are to employ a hard-limiter activation function. Furthermore, its connection weights and threshold values must be integer variables. Its basic principles are to find a set of required separating hyperplanes through a geometrical analysis of given training inputs and to determine their values of neurons. And also it requires the tedious sequential learning over all the patterns.

The MSP Term Grouping Algorithm (MTGA)is that the training patterns is represented in the boolean expression type Minimum Sum of Product (MSP) as one of digital logic synthesis method and only the terms satisfying the Unate's property as a necessary condition are combined each other. However there are some disadvantages that the not-Unate terms must be given as the additional neurons in hidden layer and the training patterns must be represented only in the MSP form.

Therefore, in the paper, a Newly Expanded and Truncated Algorithm(NETLA) which can be applied even for the case of not-Unate and other boolean expression with unusual MSP form was proposed. This NETLA decides weights and threshold values using ETL algorithm and then an optimal synthesis according to the extension principle of ETL is acquired.

Its superiority which the number of neurons in hidden layer can be reducible and the learning speed be fastened was proved through the practical application to object recognition problem such as image classification.

## 2. Preliminary

### 2.1 Structure of BNN

Assume that the binary training input patterns can be separated into the (n-1)th dimensional hyperplanes which is expressed as net function of Eq.(1) for n inputs.

$$net(x,\ T) = w_1x_1 + w_2x_2 + \ldots + w_nx_n - T = 0 \quad (1)$$

Where, $w_i$ : the connection weight between the $i$th input and the neuron

$x_i$ : the $i$th input

$T$ : the threshold value

$x$: input vector; $x = [x_1, x_2, \ldots x_n]'$

In this case, the set for training inputs must be linearly separable(LS), and the (n-1)th dimensional hyperplanes can be established by n inputs with hard-limiter activation function as Eq.(2).

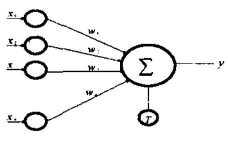


Fig. 1 The Structure of BNN

$$y = \begin{cases} 1 : \sum_{i=1}^{n} w_i x_i - T \geq 0 \\ 0 : otherwise \end{cases} \quad (2)$$

where, $y$ is output of neuron.

### 2.2 Decision of weight value and Threshold value

To decide the weights and threshold value, we can use the Reference Hypersphere(RHS) method in the ETL algorithm which can be determined from all the true and false patterns directly. RHS method used in reference hypersphere and separating hypersphere. RHS means separating hyperspheres which enclose all the training patterns and include only the true patterns. Hyperplanes are obtained by two intersecting hyperspheres defined as RHS and separating ones.

By using the geometrical ETL method, we can found the necessary hyperplanes which training inputs can be partitioned into the $n$th dimensional unit hypercube with the center, $(1/2, 1/2, \ldots, 1/2)$ and its radius $\sqrt{n}/2$, the equation for the RHS is written as Eq.(3).

$$(x_1 - \frac{1}{2})^2 + (x_2 - \frac{1}{2})^2 + \ldots + (x_n - \frac{1}{2})^2 = \frac{n}{4} \quad (3)$$

And the separating hypersphere with its center, $c_i/c_0$ and radius, $R$, inside the RHS is assumed as Eq. (4).

$$(x_1 - \frac{c_1}{c_0})^2 + (x_2 - \frac{c_2}{c_0})^2 + \ldots (x_n - \frac{c_n}{c_0})^2 = R^2 \quad (4)$$

Therefore, we can conclude that the two hyperspheres intersect each other. When these two $n$th dimensional hyperspheres intersect, the $n-1$th dimensional hyperplanes can be found. The equation describing the hyperplanes is obtained by subtracting Eq.(2) from Eq.(3) as

$$(2c_1 - c_0)x_1 + (2c_2 - c_0)x_2 + \ldots + (2c_n - c_0)x_n - T = 0 \quad (5)$$

where, $c$: center of separating hypersphere and, $c = (c_1/c_0, c_2/c_0, \cdots, c_n/c_0)$

After multiplying Eq.(5) by $c_0$

$$\sum_{i=1}^{n} w_i x_i - T = 0$$

where, $w_i = c_0 - 2c_i$

Therefore, the resultant weight is calculated as $w_i = c_0 - 2c_i$ and its threshold value is given as

$$T = R^2 c_0 - \sum_{i=1}^{n} c_i^2 / c_0 \quad (6)$$

### 2.3 Principle of Extension in the ETL algorithm

Principle of Extension in the ETL algorithm is introduced using one simple example.

Consider the given function $f(x_1, x_2, x_3)$ which has the 6-training inputs; {000, 010, 011, 111} defined as '1', {001, 100} as '0' and {101, 110} as don't care. And let $f_{min}$ be the minimum value of $\sum_{i=1}^{n}(2c_i - c_0)v_i^i$ among the rest vertices in SITV(set of included true vertices), and $f_{max}$ be the maximum of $\sum_{i=1}^{n}(2c_i - c_0)v_i^i$ among all the vertices. $f_{min} > f_{max}$, there exists a separating hyperplane which is

$$(2c_1 - c_0)x_1 + (2c_2 - c_0)x_2 + \ldots + (2c_n - c_0)x_n - T = 0$$

where, $T = \left[\frac{f_{min} + f_{max}}{2}\right]$ and $[x]$ : the smallest greater one than or equal to $x$. If $f_{min} \leq f_{max}$ there does not exist a separating hyperplane. Thus the trial vertex is removed from SITV. For example,

$f_{min} = \min[-3x_1 + x_2 - x_3]$ for SITV {000, 010, 011}, thus $f_{min} = 0 \cdot f_{max}[-3x_1 + x_2 - x_3]$ for vertices {001, 100 111}, thus $f_{max} = -1$. Since $f_{min} > f_{max}$ and $T = 0$, the hyperplane $-3x_1 + x_2 - x_3$ separates the remaining vertices {000, 010, 011} from $f_{min}$ the rest vertices. Since SITV of the first neuron includes only {000, 010, 011}, the remaining vertices are converted to expand into the first hypersphere. They are the false vertices as {001, 100} and are converted into true vertices. And the remaining true vertex {111} is converted into a false vertex. Choose one true vertex, say {001} and test if the new can be added to SITV. It turns out that SITV includes all currently declared true vertices {000, 010, 011, 001, 100}. Therefore, the algorithm is converged finding two separating hyperplanes, that is, two required neurons in hidden layer.

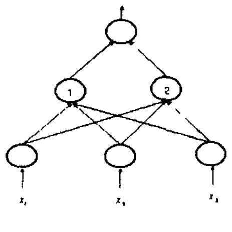

Fig. 2 The structure of a three layer binary neural network for given example.

Table 1. Weight value and threshold value in hidden layer for given example

| Input | Desired output | Hidden layer | | Output neuron |
|---|---|---|---|---|
| | | 1st neuron | 2nd neuron | |
| 000, 010, 011 | 1 | 1 | 1 | 1 |
| 001, 100 | 0 | 0 | 1 | 0 |
| 111 | 1 | 0 | 0 | 1 |

| Layer | Unit | Weight | | | Threshold |
|---|---|---|---|---|---|
| Hidden | 1 | -6 | 2 | -2 | -1 |
| | 2 | -3 | -1 | -1 | -4 |

### 2.4 Problem of synthetic image having four class

This is problem of image classification 32X32 grayscale image. Quantization result for points of 256 in the each class become output of 4 and input of 10. Each algorithm apply to this problem.

Fig. 3 The synthesis image having four class

#### 2.4.1 Synthesis result for MTGA

MTGA is synthesis result of neurons in hidden layer as Fig. 4. And the weights and threshold value in hidden layer are determined as the Table 2.

$$f_1 = x'_1x_6x_8 + x'_2x_6x_8 + x'_3x_6x_8 + x_6x_7x_8 + x'_1x'_2x'_3x_6x_7$$

$$f_2 = x_3x'_6x_7x_8 + x_2x'_6x_7x_8 + x_1x'_6x_7x_8 + x_3x_6x_7x'_8 + x_2x_6x_7x'_8 + x_1x_6x_7x'_8 + x_1x_2x_3x_6x'_7$$

$$f_3 = x'_1x'_2x'_3x'_6x_7 + x'_1x'_6x'_7x_8 + x'_2x'_6x'_7x_8 + x'_3x'_6x'_7x_8 + x'_1x_6x'_7x'_8 + x'_2x_6x'_7x'_8 + x_3x_6x'_7x'_8$$

$$f_4 = x'_6x'_7x'_8 + x_3x'_6x'_8 + x_2x'_6x'_8 + x_1x'_6x'_8 + x_1x_2x_3x'_6x'_7$$

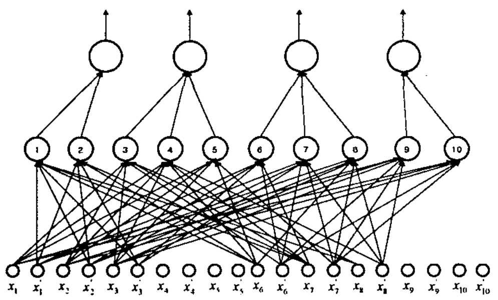

Fig. 4 The structure of three layer binary neural network for MTGA

Table 2. Weight value and threshold value in hidden layer

| Layer | | Unit | Weight | | | | | | | | | | Threshold |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hidden | 1 | 1 | -4 | -4 | -4 | 0 | 0 | 60 | 4 | 60 | 0 | 0 | 110 |
| | | 2 | -8 | -8 | -8 | 0 | 0 | 8 | 8 | 0 | 0 | 0 | 12 |
| | 2 | 3 | 4 | 4 | 4 | 0 | 0 | -28 | 28 | 28 | 0 | 0 | 58 |
| | | 4 | 4 | 4 | 4 | 0 | 0 | 28 | 28 | -28 | 0 | 0 | 58 |
| | | 5 | 8 | 8 | 8 | 0 | 0 | 8 | -8 | 0 | 0 | 0 | 28 |
| | 3 | 6 | -8 | -8 | -8 | 0 | 0 | -8 | 8 | 0 | 0 | 0 | 4 |
| | | 7 | -4 | -4 | -4 | 0 | 0 | -28 | -28 | 28 | 0 | 0 | 18 |
| | | 8 | -4 | -4 | 4 | 0 | 0 | 28 | -28 | -28 | 0 | 0 | 22 |
| | 4 | 9 | 4 | 4 | 4 | 0 | 0 | -60 | -4 | -60 | 0 | 0 | -2 |
| | | 10 | 8 | 8 | 8 | 0 | 0 | -8 | -8 | 0 | 0 | 0 | 20 |

### 2.4.2 Synthesis result for NETLA

NETLA is synthesis result of neurons in hidden layer as Fig. 8. And the weights and threshold value in hidden layer are determined as the Table 6.

$$f_1 = (x'_1x_8 + x'_2x_8 + x'_3x_8 + x_7x_8 + x'_1x'_2x'_3x_7) \cdot x_6 + (x'_1x_8 + x'_2x_8 + x'_3x_8 + x_7x_8 + x'_1x'_2x'_3x_7) \cdot 1$$

$$f_2 = (x_3x_7x_8 + x_2x_7x_8 + x_1x_7x_8) \cdot x'_6 + (x_3x_7x_8 + x_2x_7x_8 + x_1x_7x_8) \cdot 1 + (x_3x_7x'_8 + x_2x_7x'_8 + x_1x_7x'_8 + x_1x_2x_3x'_7) \cdot x_6 + (x_3x_7x'_8 + x_2x_7x'_8 + x_1x_7x'_8 + x_1x_2x_3x'_7) \cdot 1$$

$$f_3 = (x'_1x'_2x'_3x_7 + x'_1x'_7x_8 + x'_2x'_7x_8 + x'_3x'_7x_8) \cdot x'_6 + (x'_1x'_2x'_3x_7 + x'_1x'_7x_8 + x'_2x'_7x_8 + x'_3x'_7x_8) \cdot 1 + (x'_1x'_7x'_8 + x'_2x'_7x'_8 + x_3x'_7x'_8) \cdot x_6 + (x'_1x'_7x'_8 + x'_2x'_7x'_8 + x_3x'_7x'_8) \cdot 1$$

$$f_4 = (x'_7x'_8 + x_3x'_8 + x_2x'_8 + x_1x'_8 + x_1x_2x_3x'_7) \cdot x'_6 + (x'_7x'_8 + x_3x'_8 + x_2x'_8 + x_1x'_8 + x_1x_2x_3x'_7) \cdot 1$$

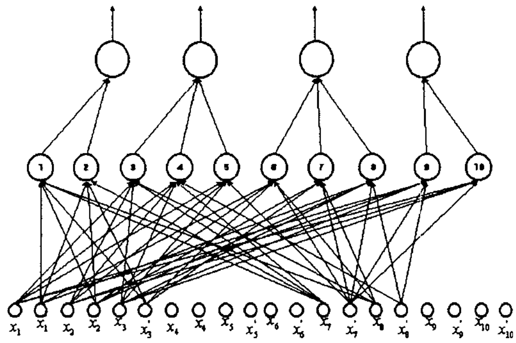

Fig. 5 The structure of a three layer neural network for NETLA

Table 3. Weight value and threshold value in hidden layer

| Layer | | Unit | Weight | | | | | | | | | | Threshold |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hidden | 1 | 1 | -4 | -4 | -4 | 0 | 0 | 0 | 60 | 4 | 0 | 0 | 50 |
| | | 2 | -8 | -8 | -8 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 20 |
| | 2 | 3 | 4 | 4 | 4 | 0 | 0 | 0 | -28 | 28 | 0 | 0 | -6 |
| | | 4 | 4 | 4 | 4 | 0 | 0 | 0 | 28 | 28 | 0 | 0 | 50 |
| | | 5 | 8 | 8 | 8 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 36 |
| | 3 | 6 | -8 | -8 | -8 | 0 | 0 | 0 | -8 | 8 | 0 | 0 | 4 |
| | | 7 | 0 | -4 | -4 | 0 | 0 | 0 | -28 | -28 | 0 | 0 | 6 |
| | | 8 | -4 | -4 | 4 | 0 | 0 | 0 | -28 | -28 | 0 | 0 | -6 |
| | 4 | 9 | 4 | 4 | 4 | 0 | 0 | 0 | -60 | -4 | 0 | 0 | 70 |
| | | 10 | 8 | 8 | 8 | 0 | 0 | 0 | -8 | 0 | 0 | 0 | 28 |

## 3. Conclusion

Because the NETLA proposed in this paper can calculate the weights and threshold from all the true and false patterns directly, it dose not require an iterative learning. And also, this algorithm has an ability to remove the unsatisfactory troubles for the functions without the Unate's property and the impossible one to describe in MSP type. The proposed algorithm. We showed that when all the true and false patterns are only given, the connection weights and the threshold values can be immediately determined by an optimal synthesis method of the NETLA without any tedious learning. Furthermore, the number of the required neurons in hidden layer can be reduced and the fast learning of BNN can be realized. The superiority of this NETLA to other algorithms was proved by the approximation problem of one circular region and image synthesis having four class. Consequently, .NETLA proposed in this paper show that decrease to neuron number in hidden layer and decrease to node number from input layer to hidden layer. NETLA proposed in this paper is proved to optimal synthesis method for binary neural network.

## Reference

[1] P. L. Bartlett and T. Downs, " Using random weight to train multilayer network of hard-limiting units," IEEE Trans. Neural Networks, pp. 202-210, 1992

[2] M. L. Brady, R. Rayhavan, and J. Slawny, " Backpropagation fails to separate where perceptrons succeed," IEEE Trans. Circuits Systems., pp. 665-674, 1989

[3] S. Park, J. H. Kim, and H. Chang, " A learning algorithm for discrete multilayer perceptron," in Proc. Int. Sysp. Circuit Systems, Singapore, 1991

[4] J. H. Kim, S. K. Park, Han, H. Oh, and M. S. Han, " The geometrical learning of binary neural network," IEEE Trans. Neural Networks, vol. 6.no. 1. pp.237-247, 1995

[5] Mitchell P. Marcus, " Switching Circuits for Engineerings," Prentice Hall, 1990